1 For AF

Attorney's Docket No.: 5220.P002X

Patent

In re the Application of: David D. Faraldo II

APPELLANT'S BRIEF UNDER 37 C.F.R. § 1.192

Application No.: 10/016,117

Filed: October 30, 2001

For: METHOD OF AND APPARATUS FOR NOTIFICATION OF STATE CHANGES IN A MONITORED SYSTEM

Mail Stop Appeal Brief - Patents
Commissioner for Patents
PO Box 1450
Alexandria, Virginia 22313-1450

SIR: Transmitted herewith is an **Appellant's Brief Under 37 C.F.R. § 1.192** for the above application.

X **Reply Brief Under 37 C.F.R. § 1.192** is enclosed. (55 pages total)
____ A check for $0.00 is attached for processing fees under 37 C.F.R. § 1.17 (f).
____ A check in the amount of $_____ is attached for presentation of additional claim(s).
____ Applicant(s) hereby Petition(s) for an Extension of Time of _____ month(s) pursuant to 37 C.F.R. § 1.136(a).
____ A check for _____ is attached for processing fees under 37 C.F.R. § 1.136 (a).
____ Please charge my Deposit Account No. 02-2666 the amount of $_____.
A duplicate copy of this sheet is enclosed.
X The Commissioner of Patents and Trademarks is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account No. 022666 **(a duplicate copy of this sheet is enclosed)**:
  X Any additional filing fees required under 37 C.F.R. § 1.16 for presentation of extra claims.
  X Any extension or petition fees under 37 C.F.R. § 1.17.

BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP

Date: June 30, 2008

Daniel E. Ovanezian

Reg. No. 41,236

1279 Oakmead Parkway
Sunnyvale, CA 94085-4040
(408) 720-8300

FIRST CLASS CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage in an envelope addressed to Mail Stop Appeal Brief - Patents, Commissioner for Patents, PO Box 1450, Alexandria, Virginia 22313-1450.

on 06/30/2008 Judy L. Steinkraus
Date of Deposit Name of Person Mailing Correspondence

Signature 06/30/08
Date

Customer No. 066701

Attorney's Docket No.: 5220.P002X

Patent

In re the Application of: David D. Faraldo II

APPELLANT'S BRIEF UNDER 37 C.F.R. § 1.192

Application No.: 10/016,117

Filed: October 30, 2001

For: METHOD OF AND APPARATUS FOR NOTIFICATION OF STATE CHANGES IN A MONITORED SYSTEM

Mail Stop Appeal Brief - Patents
Commissioner for Patents
PO Box 1450
Alexandria, Virginia 22313-1450

SIR: Transmitted herewith is an **Appellant's Brief Under 37 C.F.R. § 1.192** for the above application.

__X__ **Reply Brief Under 37 C.F.R. § 1.192** is enclosed. (55 pages total)
_____ A check for $0.00 is attached for processing fees under 37 C.F.R. § 1.17 (f).
_____ A check in the amount of $_____ is attached for presentation of additional claim(s).
_____ Applicant(s) hereby Petition(s) for an Extension of Time of _____ month(s) pursuant to 37 C.F.R. § 1.136(a).
_____ A check for _____ is attached for processing fees under 37 C.F.R. § 1.136 (a).
_____ Please charge my Deposit Account No. 02-2666 the amount of $_____.
A duplicate copy of this sheet is enclosed.
X_____ The Commissioner of Patents and Trademarks is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account No. 022666 **(a duplicate copy of this sheet is enclosed)**:
__X__ Any additional filing fees required under 37 C.F.R. § 1.16 for presentation of extra claims.
__X__ Any extension or petition fees under 37 C.F.R. § 1.17.

BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP

Date: June 30, 2008

Daniel E. Ovanezian

Reg. No. 41,236

1279 Oakmead Parkway
Sunnyvale, CA 94085-4040
(408) 720-8300

FIRST CLASS CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage in an envelope addressed to Mail Stop Appeal Brief - Patents, Commissioner for Patents, PO Box 1450, Alexandria, Virginia 22313-1450.

on 06/30/2008 Judy L. Steinkraus
Date of Deposit Name of Person Mailing Correspondence

Signature 06/30/08
Date

Customer No. 066701

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In Re Application of:

David D. Faraldo II

Application No.: 10/016,117

Filed: October 30, 2001

For: METHOD OF AND APPARATUS FOR NOTIFICATION OF STATE CHANGES IN A MONITORED SYSTEM

Examiner: Taylor, Nicholas R.

Art Unit: 2141

Confirmation Number: 7950

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage in an envelope addressed to Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.
on 06/30/2008
Date of Deposit

Judy L. Steinkraus
Name of Person Mailing Correspondence

Signature 06/30/08
Date

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

REPLY BRIEF

Pursuant to 37 C.F.R. § 1.192, Appellants submit the following Reply Brief for consideration by the Board of Patent Appeals and Interferences (hereinafter "Board"). Please charge any additional amounts due or credit any overpayment to Deposit Account No. 02-2666.

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

The real party in interest is the assignee of the full interest of the invention, Red Hat, Inc., of 1801 Varsity Drive, Raleigh, NC, 27606.

II. RELATED APPEALS AND INTERFERENCES

To the best of Appellants' knowledge, there are no appeals or interferences related to the present appeal that will directly affect, be directly affected by, or have a bearing on the Board's decision in the instant appeal.

III. STATUS OF CLAIMS

Claims 1, 2, 7-10, 15-18, 23-26, 29 and 41-44 are currently pending in the above-referenced application. Claims 1, 2, 7-10, 15-18, 23-26, 29 and 41-44 were rejected in the Final Office Action mailed on October 16, 2007, and are presented for appeal. Claims 3-6, 11-14, 19-22, 27-28 and 30-40 are canceled. A copy of claims 1-44 as they stand on appeal are set forth in Appendix A.

IV. STATUS OF AMENDMENTS

No amendments were filed subsequent to the final rejection.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

Embodiments of the instant application relate to network administration. Administrating a network may include operations such as monitoring and notification of a status of a business site's infrastructures. The notification may include standard notification rules and advanced notification rules that can suspend, redirect or

automatically acknowledge standard notifications, or transmit supplemental notifications. (See Abstract).

In an exemplary implementation of independent claim 1, a method includes enabling a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy. (Specification, page 7, lines 1-7, paragraph [0025]; page 16, lines 9-23, paragraph [0054]; Figure 7, block 730). The method further includes enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840).

In claim 2, the method generates a second notification to a second person in the hierarchy based on the advanced notification rule. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840). In claim 8, the advanced notification rule is configured to preempt the standard notification rule for a temporary amount of time. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840). In claim 41, the advanced notification rule is enabled to preempt the standard notification rule while continuing monitoring for the predetermined event. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840).

In an exemplary implementation of independent claim 9, a machine readable medium has stored thereon instructions, which when executed by a processor, cause the processor to perform the actions described below. (Specification, page 6, lines 13-19, paragraph [0024]). A standard notification rule is enabled to generate a first

notification upon an occurrence of a predetermined event to a first person in a hierarchy. (Specification, page 7, lines 1-7, paragraph [0025]; page 16, lines 10-24, paragraph [0054]; Figure 7, block 730). An advanced notification rule is enabled to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840).

In claim 10, the instructions cause the processor to generate a second notification to a second person in the hierarchy based on the advanced notification rule. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840). In claim 16, the advanced notification rule is configured to preempt the standard notification rule for a temporary amount of time. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840). In claim 42, the advanced notification rule is enabled to preempt the standard notification rule while continuing monitoring for the predetermined event. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840).

In an exemplary implementation of independent claim 17, an apparatus includes a means for enabling a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy. (Specification, page 7, lines 1-7, paragraph [0025]; page 16, lines 3-24, paragraphs [0053] - [0054]; Figure 7, block 730; Figure 5, block 580). The means for enabling the standard notification rule to generate the first notification may include a server (e.g., notification

server 570) and/or a gateway (e.g., notification gateway 580). (Specification, page 16, lines 3-9, paragraph [0053]; page 17, lines 1-8, paragraph [0055]; Figure 5). The standard notification rule can be sent to the first person in the hierarchy using a communication channel that sends communications to, for example, a pager, telephone, voicemail system, email system, etc. (Specification, page 16, lines 10-24, paragraph [0054]). The apparatus further includes a means for enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated. (Specification, page 16, lines 3-24, paragraphs [0053] - [0054]; page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840). The means for enabling the advanced notification rule to preempt the standard notification rule may include a server (e.g., notification server 570) and/or a gateway (e.g., notification gateway 580). (Specification, page 16, lines 3-9, paragraph [0053], page 17; lines 1-8, paragraph [0055]; Figure 5).

In claim 18, the apparatus includes a means for generating a second notification to a second person in the hierarchy based on the advanced notification rule. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840). The means for generating the second notification may include a server (e.g., notification server 570) and/or a gateway (e.g., notification gateway 580). (Specification, page 16, lines 3-9, paragraph [0053]; page 17, lines 1-8, paragraph [0055]; Figure 5). In claim 24, the advanced notification rule is configured to preempt the standard notification rule for a temporary amount of time. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks

810-840). In claim 43, the advanced notification rule is enabled to preempt the standard notification rule while continuing monitoring for the predetermined event. (Specification, page 19, line 14, paragraph [0063] – page 23, line 7, paragraph [0075]; Figure 8, blocks 810-840).

In an exemplary implementation of independent claim 25, a digital processing system includes a processor configured to enable a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy. (Specification, page 7, lines 1-7, paragraph [0025]; page 9, lines 14-23, paragraph [0034]; page 16, lines 10-24, paragraph [0054]; Figure 7, block 730; Figure 3, block 302). The processor is further configured to enable an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated. (Specification, page 9, lines 14-23, paragraph [0034]; page 19, lines 14-20, paragraph [0063]; page 20, lines 14-18, paragraph [0066]; Figure 8, block 810; Figure 3, block 302). The digital processing system includes a communications device coupled to the processor to transmit the notifications. (Figure 3, blocks 325-326; page 10, lines 17-23, paragraph [0037]; page 11, lines 3-7, paragraph [0039]; page 16, lines 10-24, paragraph [0054]).

In claim 26, the communications device is configured to transmit the second notification to a second person in the hierarchy based on the advanced notification rule. (Specification, page 19, line 14, paragraph [0063] – page 23, lines 7, paragraph [0075]; Figure 8, blocks 810-840). In claim 44, the processor is configured to enable the advanced notification rule to preempt the standard notification rule while continuing

monitoring for the predetermined event. (Specification, page 19, line 14, paragraph [0063] – page 23, lines 7, paragraph [0075]; Figure 8, blocks 810-840).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The issues involved in this Appeal are as follows:

A. Whether claims 1, 7-9, 15-17, 23-25, 29 and 41-44 are anticipated by U.S. Patent No. 5,987,514 to Rangarajan ("Rangarajan").

B. Whether claims 2, 10, 18 and 26 are unpatentable over the combination of U.S. Patent No. 5,987,514 to Rangarajan ("Rangarajan") and U.S. Patent No. 5,987,514 to Graf ("Graf").

VII. ARGUMENT

A. Claims 1, 7-9, 15-17, 23-25, 29 and 41-44 are not anticipated by U.S. Patent No. 5,987,514 to Rangarajan ("Rangarajan") because Rangarajan fails to disclose each of the elements of these claims.

1. Claims 1 and 9 and associated dependent claims 2, 7, 8, 10, 15, 16, 41 and 42 are not anticipated by Rangarajan because Rangarajan fails to disclose enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated.

Appellants respectfully submit that Rangarajan does not disclose an advanced notification rule that preempts a standard notification rule. Rangarajan discloses a network manager that generates event requests and sends them to mid-level managers. The mid-level managers generate event reports and send them back to the network manager. Upon receiving the event reports, the network manager performs a signaling action (e.g., sounds an alarm). (Rangarajan, col. 5, lines 39-56). The examiner has interpreted the "signaling action" of Rangarajan as a notification. (Office Action, 10/16/2007, page 2). As the examiner has pointed out, the "signaling action" may include sounding an alarm, sending an e-mail, or providing visual displays. (Rangarajan, col. 1, lines 36-40). However, Rangarajan does not disclose any rules that determine under what circumstances particular signaling actions should be performed. Therefore, Rangarajan **fails to explicitly disclose any notification rules**. In contrast to Rangarajan, claims 1 and 9 include a **standard notification rule** to generate a first notification upon an occurrence of a predetermined event to a first

person in a hierarchy **and an advanced notification rule** to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence of the predetermined event.

In the office action of October 16, 2007, the Examiner cited col. 9, lines 19-58 and col. 5, lines 39-63 as disclosing, "enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated." (Office Action, 10/17/2007, page 4). However, the Office Action failed to provide any analysis of how or why the claims were asserted to be anticipated by the disclosure of Rangarajan. In the Examiner's Answer, the Examiner further cited col. 6, lines 59-63 and col. 7, lines 1-59, and attempted to clarify what language of Rangarajan is purported to disclose a notification rule and an advanced notification rule, stating, "the notification rule is equivalent to what is contained in Rangarajan's event request record." However, each event request record includes numerous components. The Examiner has failed to explicitly point to any particular component of the event request records of Rangarajan as being the same as a notification rule.

The Examiner's Response states that, "[m]any actions can be taken ..., including stopping all rules from processing (via preemption), sending a request to another device, repeating a request over a different pathway, sending a notification, or taking no action at all (col. 5, line 51 to col. 6, line 29)." (Examiner's Response, page 8). It appears that Examiner is purporting that the actions disclosed by Rangarajan are the same as both the notification rule and the advanced notification rule recited in claims 1

and 9. Appellants respectfully disagree, and submit that such an interpretation is inapposite.

Rangarajan discloses four different actions that may be performed when predetermined conditions are met. These actions consist of (1) sending a stop request, (2) sending a start request, (3) sending an event request to an alternate mid-level manager, or (4) taking no action. (Rangarajan, col. 5, line 56 to col. 6, line 29). Each action is associated with a separate rule that designates when to perform the action. However, such rule based actions as described by Rangarajan do not include sending a notification. Rangarajan explicitly limits the actions that can be performed by the network manager in response to certain conditions being met to the four disclosed actions, stating, "[i]f none of the fields (e.g., Start, Sop and Alternate Proxy fields) are filled in or checked off, the network manager takes no action." (Rangarajan, col. 9, lines 9-11). Rangarajan does not disclose a "notification field," or any fields that might otherwise include a notification rule. Moreover, generating a first notification to a first person in a hierarchy is not among the four possible actions disclosed by Rangarajan. Additionally, Rangarajan does not disclose any rules that specify when, if, or to whom to send a notification. In contrast, claims 1 and 9 recite, "enabling a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy."

The Examiner's Response cites col. 2, lines 19-32 of Rangarajan as disclosing teaching a network management system that generates notifications in response to network events based on different rules. Appellants respectfully disagree with Examiner's reading of Rangarajan. Rangarajan states that, "[i]n response to the event

report, the network manager notifies a network administrator of the event by performing a signaling action such as sounding an alarm." (Rangarajan, col. 5, lines 53-56). However, Rangarajan does not disclose that the type of signaling action performed is based upon any rule, or that different signaling actions may be performed for different event reports. For example, Rangarajan fails to disclose that a first signaling action is performed when first conditions are met, or that a second signaling action is performed when second conditions are met. In Rangarajan, no notification rules are disclosed, and the same signaling action is performed regardless of the content of the event report.

The Examiner's Response also cites col. 5, lines 50-63 and col. 8, lines 40-49 as disclosing that the network manager of Rangarajan can take an action based on predefined responses for an event, such as sending a notification or sounding an alarm. Again, Appellants respectfully disagree with examiner's reading of Rangarajan. As discussed above, Rangarajan lists four different types of actions that can be performed if a monitored attribute satisfies a condition: (1) send a stop request, (2) send a start request, (3) send an event request to an alternate mid-level manager, or (4) take no action. (Rangarajan, col. 5, line 56 to col. 6, line 29). Rangarajan does not disclose performing any notification action if a monitored attribute satisfies a condition.

Even, for the sake of argument, if Rangarajan were to be read in an overly broad sense as inherently including a standard notification rule, such a reading would not include an advanced notification rule capable of preempting the standard notification rule. Accordingly, Rangarajan fails to disclose all of the features of independent claims 1 and 9.

Examiner's Response states that a stop event can be used in an advanced notification rule to preempt standard notification rules from being generated upon the occurrence of a predetermined event such that a first notification is not generated. Appellants respectfully disagree, and submit that such an interpretation is inapposite.

A "stop" event request as described by Rangarajan is not the same as an advanced notification rule claimed in claims 1 and 9. Rangarajan defines an event request as a request that directs a mid-level manager to poll a device during a prescribed interval to ascertain an attribute of the device against one or more conditions, and a "stop" event request, in particular, as an event request that commands the mid-level manager to stop polling (and therefore to stop generating event reports) for the attribute. (Rangarajan, col. 3, lines 32-34; col. 8, lines 43-47). Stop event requests are issued by the network manager to a mid-level manager in response to receiving an event report if certain attribute conditions are satisfied. Signaling actions are also generated by the network manager in response to receiving the event report, regardless of attribute conditions. (Rangarajan, col. 5, line 53-63). If, as the Examiner seems to suggest, the "stop" event request were to preempt a notification rule to suspend the signaling action from occurring, then no signaling event would ever occur for the event (given that the stop event request ceases generation of event reports, and signaling actions occur upon receipt of event reports). (See Rangarajan, col. 9, lines 51-53). Therefore, no system administrators would ever be notified of the condition that caused the original event report. This would result in an effectively non-functional system. Therefore, the "stop" event request of Rangarajan can not preempt any rules that might cause the signaling actions, and can not be a notification rule, advanced or otherwise.

The Examiner's response states that appellants are using an interpretation of the phrase, "occurrence of a predetermined event" that is too limiting. Specifically, Examiner's Response states that "occurrence of a predetermined event" should include a device failure that lasts for a lengthy period of time, and that under such an interpretation the stop event request of Rangarajan would act as an advanced notification rule. Appellants respectfully submit that under Examiner's reading, in which predetermined events last for a lengthy period of time, Rangarajan still fails to disclose an advanced notification rule that preempts a standard notification rule, as recited in claims 1 and 9.

The Examiner's Response provides an example to illustrate Examiner's assertion that a stop event request is the same as an advanced notification rule. Specifically, the Examiner's Response states:

> A first event request record (i.e., the standard notification rule) would start at 3:00 p.m. and would send a notification if the device failed (see start time field, threshold one, and relation one of column 7). A second event request record (i.e., the advanced notification rule) would start earlier at 1:00 p.m. and would stop all polling if the device failed in the CPU usage is high (see start time field, both thresholds, and both thresholds of column 7). If at 2:00 PM a device had high CPU usage and failed, the advanced notification rule will trigger and send a stop event.
>
> Such a stop event would preempt the standard notification rule by suspending the first notification rule from being generated upon the occurrence such that the first notification rule is not generated. That is, when 3:00 PM arrived the standard notification rule would not poll because of the stop event and the first notification would not be generated.

(Examiner's Response, page 9).

Appellants respectfully disagree with Examiner's characterization of Rangarajan and submit that Examiner's example does not accurately reflect what is disclosed in Rangarajan. In the Examiner's example, a stop event request disables all subsequent

event requests from being performed. However, the stop request is described by Rangarajan in relation to a single specific event request. Each event request includes parameters that determine when, how often, and at what intervals a device will be polled in response to the event request. (Rangarajan, col. 7, lines 19-34). Unless a stop event request is issued for an event request, that event request will cause a low-level agent to be continually polled for the designated time period. If the stop request field is checked off however, polling of an attribute is stopped for that event request when the monitored attribute satisfies a specified condition. (Rangarajan, col. 7, lines 60-62, col. 8, lines 43-48). As disclosed by Rangarajan, this reduces unnecessary network traffic.

Rangarajan further states that, "once the alternate proxy is delegated, the previous mid-level manager is disengaged by sending a stop event request to the previous mid-level manager." If the stop event request were to stop polling of an attribute for all event requests, then an administrator would have to manually re-enable an attribute every time an alternate proxy is delegated. This would introduce an unnecessary burden to the administrator. Therefore, stop event requests only disable polling of an attribute for a specific event request. Accordingly, if the stop event request were to preempt a notification to suspend a signaling action from occurring, no signaling action would ever occur for the event. As previously stated by Appellants, this would result in an effectively non-functional system.

Referring to the Examiner's example, if at 2:00PM a stop event request is triggered, this would stop polling of the device by only the second event request. The first event request would still begin polling the device at 3:00 PM. Presumably, the first event

request would include its own conditions for sending a stop event request. Therefore, if the device has failed as in the Examiner's answer, the first event request would poll the device only briefly, trigger a stop event request for the first event request, and cease polling the device. The stop event request of the second event record would not preempt the first event record from polling the device, and therefore would not preempt a signaling action generated by the first even record. Accordingly, the stop event request of Rangarajan is not the same as an advanced notification rule.

The Office Action of October 16, 2007 also cited col. 5, lines 56-63 of Rangarajan as disclosing, "enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated." The cited passage is reproduced below:

> The council procedure 70 includes an object-oriented graphical user interface (GUI) for modifying the event request and attributes records in the runtime database 62. The GUI can be derived from open windows 3.1 or later or any other library of classes for GUIs. To modify a record, the console procedure 70 displays the fields for an event request (the fields of an event request record and one or more attributes records can be combined into a single display) and allows the network administrator to fill in or change the fields. The modified records are saved, and the console procedure 70 is restarted.
>
> Reference is now made to FIG. 6, which shows the steps performed by a network administrator while using the console procedure 70. First, the event dispatcher 68 is run in the background (step 200) and then the console procedure 70 is executed (step 202). Upon execution, the console procedure 70 registers with the event dispatcher 68, informing the event dispatcher 68 to forward event reports 78 to it.
>
> If any of the records in the runtime database 62 need to be modified (step 204), the network administrator modifies and saves the records in the runtime database 62 (step 206). The console procedure 70 begins firing event requests 82 at their scheduled start times.

> If an event is generated (step 208), the network administrator can view the corresponding event reports 78 via the console procedure 70 (step 210). The event report 78 indicates the attribute and conditions for which the event was generated, the course of action taken by the network manager 48, and the results (if any) from actions taken.
>
> If the event report 78 indicates that the device was down because CPU usage was too high, or because a router on the path was not operational, the network administrator can take the appropriate actions (step 212). If polling of the device attribute has been stopped, no remedial further event reports 78 will be generated for the device.
>
> Thus disclosed is an invention that reduces network management traffic and performs troubleshooting automatically and conveniently from a remote location. The invention greatly reduces the burden of managing a network.

(Rangarajan, col. 5, lines 56-63).

Although it is unclear what language of the cited passage of Rangarajan the Examiner is purporting to disclose an advanced notification rule and a first notification, it appears that the Examiner may be attempting to interpret Rangarajan's disclosure of an event request as an advanced notification rule and Rangarajan's disclosure of an event report as the first notification. It is respectfully submitted that such an interpretation is inapposite.

An event request as described by Rangarajan is not the same as an advanced notification rule. Nor is an event report as described by Rangarajan the same as a notification. Rangarajan defines an event request as a message sent from a network manager to a mid-level manager that directs the mid-level manager to poll a device during a prescribed interval to ascertain an attribute of the device against one or more conditions, and an event report as a message forwarded to the network manager by the mid-level manager when the one or more conditions occur. (Rangarajan, col. 3, lines 32-37). The event request does not include any rules that identify when or how to notify

a system administrator or other user when a condition occurs. Nor does the event report include any notification to a system administrator or other user that the condition has occurred. Such a notification is instead accomplished by a signaling action performed by the network manager, the signaling action being distinct from the event request and the event report. (Rangarajan, col. 5, lines 53-56). However, Rangarajan fails to disclose any rules that control when or how to perform a signaling action. In contrast, claims 1 and 9 recite, "enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated."

Rangarajan fails to disclose all of the features of claims 1 and 9. Accordingly, independent claim 1 and dependent claims 2, 7, 8 and 41, and independent claim 9 and dependent claims 10, 15, 16 and 42 are not anticipated by Rangarajan.

2. Claims 1 and 9 and associated dependent claims 2, 7, 8, 10, 15, 16, 41 and 42 are not anticipated by Rangarajan because Rangarajan fails to disclose enabling a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy.

Appellants respectfully submit that Rangarajan does not disclose a notification rule that generates a first notification upon an occurrence of a predetermined event to a first person in a hierarchy. Rangarajan discloses a network manager that performs a signaling action (e.g., sounds an alarm) upon receiving event reports. (Rangarajan, col. 5, lines 39-56). The signaling action notifies a network administrator of the event. (Rangarajan, col. 5, lines 53-56). However, Rangarajan does not disclose multiple

different persons who can be notified of an event. Nor does Rangarajan disclose a hierarchy of persons to notify of the event. Rangarajan also does not disclose any rules that determine under what circumstances a particular person should be notified of an event. In contrast, claims 1 and 9 recite, "enabling a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy."

Rangarajan fails to disclose all of the features of claims 1 and 9. Accordingly, independent claim 1 and dependent claims 2, 7, 8 and 41, and independent claim 9 and dependent claims 10, 15, 16 and 42 are not anticipated by Rangarajan.

3. Claim 17 and associated dependent claims 18, 23, 24 and 43 are not anticipated by Rangarajan because Rangarajan fails to disclose means for enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated.

Appellants respectfully submit that Rangarajan does not disclose means for enabling an advanced notification rule to preempt a standard notification rule. Rangarajan discloses a network manager that generates event requests and sends them to mid-level managers. The mid-level managers generate event reports and send them back to the network manager. Upon receiving the event reports, the network manager performs a signaling action (e.g., sounds an alarm). (Rangarajan, col. 5, lines 39-56). The examiner has interpreted the "signaling action" of Rangarajan as a notification. (Office Action, 10/16/2007, page 2). As the examiner has pointed out, the

"signaling action" may include sounding an alarm, sending an e-mail, or providing visual displays. (Rangarajan, col. 1, lines 36-40). However, Rangarajan does not disclose any rules that determine under what circumstances particular signaling actions should be performed. Therefore, Rangarajan **fails to explicitly disclose any notification rules**. In contrast to Rangarajan, claim 17 includes means for enabling a **standard notification rule** to generate a first notification upon an occurrence of a predetermined event and means for enabling **an advanced notification rule** to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence of the predetermined event.

In the office action of October 16, 2007, the Examiner cited col. 9, lines 19-58 and col. 5, lines 39-63 as disclosing, "enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated." (Office Action, 10/17/2007, page 4). However, the Office Action failed to provide any analysis of how or why the claims were asserted to be anticipated by the disclosure of Rangarajan. In the Examiner's Answer, the Examiner further cited col. 6, lines 59-63 and col. 7, lines 1-59, and attempted to clarify what language of Rangarajan is purported to disclose a notification rule and an advanced notification rule, stating, "the notification rule is equivalent to what is contained in Rangarajan's event request record." However, each event request record includes numerous components. The Examiner has failed to explicitly point to any particular component of the event request records of Rangarajan as being the same as a notification rule.

The Examiner's Response states that, "[m]any actions can be taken ..., including stopping all rules from processing (via preemption), sending a request to another device, repeating a request over a different pathway, sending a notification, or taking no action at all (col. 5, line 51 to col. 6, line 29)." (Examiner's Response, page 8). It appears that Examiner is purporting that the actions disclosed by Rangarajan are the same as both the notification rule and the advanced notification rule recited in claim 17. Appellants respectfully disagree, and submit that such an interpretation is inapposite.

Rangarajan discloses four different actions that may be performed when predetermined conditions are met. These actions consist of (1) sending a stop request, (2) sending a start request, (3) sending an event request to an alternate mid-level manager, or (4) taking no action. (Rangarajan, col. 5, line 56 to col. 6, line 29). Each action is associated with a separate rule that designates when to perform the action. However, such rule based actions as described by Rangarajan do not include sending a notification. Rangarajan explicitly limits the actions that can be performed by the network manager in response to certain conditions being met to the four disclosed actions, stating, "[i]f none of the fields (e.g., Start, Sop and Alternate Proxy fields) are filled in or checked off, the network manager takes no action." (Rangarajan, col. 9, lines 9-11). Rangarajan does not disclose a "notification field," or any fields that might otherwise include a notification rule. Moreover, generating a first notification to a first person in a hierarchy is not among the four possible actions disclosed by Rangarajan. Additionally, Rangarajan does not disclose any rules that specify when, if, or to whom to send a notification. In contrast, claim 17 recites, "means for enabling a standard

notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy."

The Examiner's Response cites col. 2, lines 19-32 of Rangarajan as disclosing teaching a network management system that generates notifications in response to network events based on different rules. Appellants respectfully disagree with Examiner's reading of Rangarajan. Rangarajan states that, "[i]n response to the event report, the network manager notifies a network administrator of the event by performing a signaling action such as sounding an alarm." (Rangarajan, col. 5, lines 53-56). However, Rangarajan does not disclose that the type of signaling action performed is based upon any rule, or that different signaling actions may be performed for different event reports. For example, Rangarajan fails to disclose that a first signaling action is performed when first conditions are met, or that a second signaling action is performed when second conditions are met. In Rangarajan, no notification rules are disclosed, and the same signaling action is performed regardless of the content of the event report.

The Examiner's Response also cites col. 5, lines 50-63 and col. 8, lines 40-49 as disclosing that the network manager of Rangarajan can take an action based on predefined responses for an event, such as sending a notification or sounding an alarm. Again, Appellants respectfully disagree with examiner's reading of Rangarajan. As discussed above, Rangarajan lists four different types of actions that can be performed if a monitored attribute satisfies a condition: (1) send a stop request, (2) send a start request, (3) send an event request to an alternate mid-level manager, or (4) take no

action. (Rangarajan, col. 5, line 56 to col. 6, line 29). Rangarajan does not disclose performing any notification action if a monitored attribute satisfies a condition.

Even, for the sake of argument, if Rangarajan were to be read in an overly broad sense as inherently including a standard notification rule, such a reading would not include an advanced notification rule capable of preempting the standard notification rule. Accordingly, Rangarajan fails to disclose all of the features of independent claim 17.

Examiner's Response states that a stop event can be used in an advanced notification rule to preempt standard notification rules from being generated upon the occurrence of a predetermined event such that a first notification is not generated. Appellants respectfully disagree, and submit that such an interpretation is inapposite.

A "stop" event request as described by Rangarajan is not the same as an advanced notification rule claimed in claim 17. Rangarajan defines an event request as a request that directs a mid-level manager to poll a device during a prescribed interval to ascertain an attribute of the device against one or more conditions, and a "stop" event request, in particular, as an event request that commands the mid-level manager to stop polling (and therefore to stop generating event reports) for the attribute. (Rangarajan, col. 3, lines 32-34; col. 8, lines 43-47). Stop event requests are issued by the network manager to a mid-level manager in response to receiving an event report if certain attribute conditions are satisfied. Signaling actions are also generated by the network manager in response to receiving the event report, regardless of attribute conditions. (Rangarajan, col. 5, line 53-63). If, as the Examiner seems to suggest, the "stop" event request were to preempt a notification rule to suspend the signaling action from

occurring, then no signaling event would ever occur for the event (given that the stop event request ceases generation of event reports, and signaling actions occur upon receipt of event reports). (See Rangarajan, col. 9, lines 51-53). Therefore, no system administrators would ever be notified of the condition that caused the original event report. This would result in an effectively non-functional system. Therefore, the "stop" event request of Rangarajan can not preempt any rules that might cause the signaling actions, and can not be a notification rule, advanced or otherwise.

The Examiner's response states that appellants are using an interpretation of the phrase, "occurrence of a predetermined event" that is too limiting. Specifically, Examiner's Response states that "occurrence of a predetermined event" should include a device failure that lasts for a lengthy period of time, and that under such an interpretation the stop event request of Rangarajan would act as an advanced notification rule. Appellants respectfully submit that under Examiner's reading, in which predetermined events last for a lengthy period of time, Rangarajan still fails to disclose an advanced notification rule that preempts a standard notification rule, as recited in claim 17.

The Examiner's Response provides an example to illustrate Examiner's assertion that a stop event request is the same as an advanced notification rule. Specifically, the Examiner's Response states:

> A first event request record (i.e., the standard notification rule) would start at 3:00 p.m. and would send a notification if the device failed (see start time field, threshold one, and relation one of column 7). A second event request record (i.e., the advanced notification rule) would start earlier at 1:00 p.m. and would stop all polling if the device failed in the CPU usage is high (see start time field, both thresholds, and both thresholds of column 7). If at

2:00 PM a device had high CPU usage and failed, the advanced notification rule will trigger and send a stop event.

Such a stop event would preempt the standard notification rule by suspending the first notification rule from being generated upon the occurrence such that the first notification rule is not generated. That is, when 3:00 PM arrived the standard notification rule would not poll because of the stop event and the first notification would not be generated.

(Examiner's Response, page 9).

Appellants respectfully disagree with Examiner's characterization of Rangarajan and submit that Examiner's example does not accurately reflect what is disclosed in Rangarajan. In the Examiner's example, a stop event request disables all subsequent event requests from being performed. However, the stop request is described by Rangarajan in relation to a single specific event request. Each event request includes parameters that determine when, how often, and at what intervals a device will be polled in response to the event request. (Rangarajan, col. 7, lines 19-34). Unless a stop event request is issued for an event request, that event request will cause a low-level agent to be continually polled for the designated time period. If the stop request field is checked off however, polling of an attribute is stopped for that event request when the monitored attribute satisfies a specified condition. (Rangarajan, col. 7, lines 60-62, col. 8, lines 43-48). As disclosed by Rangarajan, this reduces unnecessary network traffic.

Rangarajan further states that, "once the alternate proxy is delegated, the previous mid-level manager is disengaged by sending a stop event request to the previous mid-level manager." If the stop event request were to stop polling of an attribute for all event requests, then an administrator would have to manually re-enable an attribute every time an alternate proxy is delegated. This would introduce an unnecessary burden to

the administrator. Therefore, stop event requests only disable polling of an attribute for a specific event request. Accordingly, if the stop event request were to preempt a notification to suspend a signaling action from occurring, no signaling action would ever occur for the event. As previously stated by Appellants, this would result in an effectively non-functional system.

Referring to the Examiner's example, if at 2:00PM a stop event request is triggered, this would stop polling of the device by only the second event request. The first event request would still begin polling the device at 3:00 PM. Presumably, the first event request would include its own conditions for sending a stop event request. Therefore, if the device has failed as in the Examiner's answer, the first event request would poll the device only briefly, trigger a stop event request for the first event request, and cease polling the device. The stop event request of the second event record would not preempt the first event record from polling the device, and therefore would not preempt a signaling action generated by the first even record. Accordingly, the stop event request of Rangarajan is not the same as an advanced notification rule.

The Office Action of October 16, 2007 also cited col. 5, lines 56-63 of Rangarajan as disclosing, "enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated." The cited passage is reproduced below:

> The council procedure 70 includes an object-oriented graphical user interface (GUI) for modifying the event request and attributes records in the runtime database 62. The GUI can be derived from open windows 3.1 or later or any other library of classes for GUIs. To modify a record, the console procedure 70 displays the fields for an event request (the fields of an event

request record and one or more attributes records can be combined into a single display) and allows the network administrator to fill in or change the fields. The modified records are saved, and the console procedure 70 is restarted.

Reference is now made to FIG. 6, which shows the steps performed by a network administrator while using the console procedure 70. First, the event dispatcher 68 is run in the background (step 200) and then the console procedure 70 is executed (step 202). Upon execution, the console procedure 70 registers with the event dispatcher 68, informing the event dispatcher 68 to forward event reports 78 to it.

If any of the records in the runtime database 62 need to be modified (step 204), the network administrator modifies and saves the records in the runtime database 62 (step 206). The console procedure 70 begins firing event requests 82 at their scheduled start times.

If an event is generated (step 208), the network administrator can view the corresponding event reports 78 via the console procedure 70 (step 210). The event report 78 indicates the attribute and conditions for which the event was generated, the course of action taken by the network manager 48, and the results (if any) from actions taken.

If the event report 78 indicates that the device was down because CPU usage was too high, or because a router on the path was not operational, the network administrator can take the appropriate actions (step 212). If polling of the device attribute has been stopped, no remedial further event reports 78 will be generated for the device.

Thus disclosed is an invention that reduces network management traffic and performs troubleshooting automatically and conveniently from a remote location. The invention greatly reduces the burden of managing a network.

(Rangarajan, col. 5, lines 56-63).

Although it is unclear what language of the cited passage of Rangarajan the Examiner is purporting to disclose an advanced notification rule and a first notification, it appears that the Examiner may be attempting to interpret Rangarajan's disclosure of an event request as an advanced notification rule and Rangarajan's disclosure of an event report as the first notification. It is respectfully submitted that such an interpretation is inapposite.

An event request as described by Rangarajan is not the same as an advanced notification rule. Nor is an event report as described by Rangarajan the same as a notification. Rangarajan defines an event request as a message sent from a network manager to a mid-level manager that directs the mid-level manager to poll a device during a prescribed interval to ascertain an attribute of the device against one or more conditions, and an event report as a message forwarded to the network manager by the mid-level manager when the one or more conditions occur. (Rangarajan, col. 3, lines 32-37). The event request does not include any rules that identify when or how to notify a system administrator or other user when a condition occurs. Nor does the event report include any notification to a system administrator or other user that the condition has occurred. Such a notification is instead accomplished by a signaling action performed by the network manager, the signaling action being distinct from the event request and the event report. (Rangarajan, col. 5, lines 53-56). However, Rangarajan fails to disclose any rules that control when or how to perform a signaling action. In contrast, claim 17 recites, "means for enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated."

Rangarajan fails to disclose all of the features of claim 17. Accordingly, independent claim 17 and dependent claims 18, 23, 24 and 43 are not anticipated by Rangarajan.

4. Claim 17 and associated dependent claims 18, 23, 24 and 43 are not anticipated by Rangarajan because Rangarajan fails to disclose means for enabling a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy.

Appellants respectfully submit that Rangarajan does not disclose means for enabling a notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy. Rangarajan discloses a network manager that performs a signaling action (e.g., sounds an alarm) upon receiving event reports. (Rangarajan, col. 5, lines 39-56). The signaling action notifies a network administrator of the event. (Rangarajan, col. 5, lines 53-56). However, Rangarajan does not disclose multiple different persons who can be notified of an event. Nor does Rangarajan disclose a hierarchy of persons to notify of the event. Rangarajan also does not disclose any rules that determine under what circumstances a particular person should be notified of an event. In contrast, claim 17 recites, "means for enabling a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy."

Rangarajan fails to disclose all of the features of claim 17. Accordingly, independent claim 17 and dependent claims 18, 23, 24 and 43 are not anticipated by Rangarajan.

5. Claim 25 and associated dependent claims 26, 29 and 44 are not anticipated by Rangarajan because Rangarajan fails to disclose a processor configured to enable an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated.

Appellants respectfully submit that Rangarajan does not disclose a processor configured to enable an advanced notification rule to preempt a standard notification rule. Rangarajan discloses a network manager that generates event requests and sends them to mid-level managers. The mid-level managers generate event reports and send them back to the network manager. Upon receiving the event reports, the network manager performs a signaling action (e.g., sounds an alarm). (Rangarajan, col. 5, lines 39-56). The examiner has interpreted the "signaling action" of Rangarajan as a notification. (Office Action, 10/16/2007, page 2). As the examiner has pointed out, the "signaling action" may include sounding an alarm, sending an e-mail, or providing visual displays. (Rangarajan, col. 1, lines 36-40). However, Rangarajan does not disclose any rules that determine under what circumstances particular signaling actions should be performed. Therefore, Rangarajan **fails to explicitly disclose any notification rules**. In contrast to Rangarajan, claim 25 includes a processor configured to enable a **standard notification rule** to generate a first notification upon an occurrence of a predetermined event and to enable **an advanced notification rule** to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence of the predetermined event.

In the office action of October 16, 2007, the Examiner cited col. 9, lines 19-58 and col. 5, lines 39-63 as disclosing, "enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated." (Office Action, 10/17/2007, page 4). However, the Office Action failed to provide any analysis of how or why the claims were asserted to be anticipated by the disclosure of Rangarajan. In the Examiner's Answer, the Examiner further cited col. 6, lines 59-63 and col. 7, lines 1-59, and attempted to clarify what language of Rangarajan is purported to disclose a notification rule and an advanced notification rule, stating, "the notification rule is equivalent to what is contained in Rangarajan's event request record." However, each event request record includes numerous components. The Examiner has failed to explicitly point to any particular component of the event request records of Rangarajan as being the same as a notification rule.

The Examiner's Response states that, "[m]any actions can be taken ..., including stopping all rules from processing (via preemption), sending a request to another device, repeating a request over a different pathway, sending a notification, or taking no action at all (col. 5, line 51 to col. 6, line 29)." (Examiner's Response, page 8). It appears that Examiner is purporting that the actions disclosed by Rangarajan are the same as both the notification rule and the advanced notification rule recited in claim 25. Appellants respectfully disagree, and submit that such an interpretation is inapposite.

Rangarajan discloses four different actions that may be performed when predetermined conditions are met. These actions consist of (1) sending a stop request, (2) sending a start request, (3) sending an event request to an alternate mid-level

manager, or (4) taking no action. (Rangarajan, col. 5, line 56 to col. 6, line 29). Each action is associated with a separate rule that designates when to perform the action. However, such rule based actions as described by Rangarajan do not include sending a notification. Rangarajan explicitly limits the actions that can be performed by the network manager in response to certain conditions being met to the four disclosed actions, stating, "[i]f none of the fields (e.g., Start, Sop and Alternate Proxy fields) are filled in or checked off, the network manager takes no action." (Rangarajan, col. 9, lines 9-11). Rangarajan does not disclose a "notification field," or any fields that might otherwise include a notification rule. Moreover, generating a first notification to a first person in a hierarchy is not among the four possible actions disclosed by Rangarajan. Additionally, Rangarajan does not disclose any rules that specify when, if, or to whom to send a notification. In contrast, claim 25 recites, "a processor configured to enable a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy."

The Examiner's Response cites col. 2, lines 19-32 of Rangarajan as disclosing teaching a network management system that generates notifications in response to network events based on different rules. Appellants respectfully disagree with Examiner's reading of Rangarajan. Rangarajan states that, "[i]n response to the event report, the network manager notifies a network administrator of the event by performing a signaling action such as sounding an alarm." (Rangarajan, col. 5, lines 53-56). However, Rangarajan does not disclose that the type of signaling action performed is based upon any rule, or that different signaling actions may be performed for different event reports. For example, Rangarajan fails to disclose that a first signaling action is

performed when first conditions are met, or that a second signaling action is performed when second conditions are met. In Rangarajan, no notification rules are disclosed, and the same signaling action is performed regardless of the content of the event report.

The Examiner's Response also cites col. 5, lines 50-63 and col. 8, lines 40-49 as disclosing that the network manager of Rangarajan can take an action based on predefined responses for an event, such as sending a notification or sounding an alarm. Again, Appellants respectfully disagree with examiner's reading of Rangarajan. As discussed above, Rangarajan lists four different types of actions that can be performed if a monitored attribute satisfies a condition: (1) send a stop request, (2) send a start request, (3) send an event request to an alternate mid-level manager, or (4) take no action. (Rangarajan, col. 5, line 56 to col. 6, line 29). Rangarajan does not disclose performing any notification action if a monitored attribute satisfies a condition.

Even, for the sake of argument, if Rangarajan were to be read in an overly broad sense as inherently including a standard notification rule, such a reading would not include an advanced notification rule capable of preempting the standard notification rule. Accordingly, Rangarajan fails to disclose all of the features of independent claim 25.

Examiner's Response states that a stop event can be used in an advanced notification rule to preempt standard notification rules from being generated upon the occurrence of a predetermined event such that a first notification is not generated. Appellants respectfully disagree, and submit that such an interpretation is inapposite.

A "stop" event request as described by Rangarajan is not the same as an advanced notification rule claimed in claim 25. Rangarajan defines an event request as a request that directs a mid-level manager to poll a device during a prescribed interval to ascertain an attribute of the device against one or more conditions, and a "stop" event request, in particular, as an event request that commands the mid-level manager to stop polling (and therefore to stop generating event reports) for the attribute. (Rangarajan, col. 3, lines 32-34; col. 8, lines 43-47). Stop event requests are issued by the network manager to a mid-level manager in response to receiving an event report if certain attribute conditions are satisfied. Signaling actions are also generated by the network manager in response to receiving the event report, regardless of attribute conditions. (Rangarajan, col. 5, line 53-63). If, as the Examiner seems to suggest, the "stop" event request were to preempt a notification rule to suspend the signaling action from occurring, then no signaling event would ever occur for the event (given that the stop event request ceases generation of event reports, and signaling actions occur upon receipt of event reports). (See Rangarajan, col. 9, lines 51-53). Therefore, no system administrators would ever be notified of the condition that caused the original event report. This would result in an effectively non-functional system. Therefore, the "stop" event request of Rangarajan can not preempt any rules that might cause the signaling actions, and can not be a notification rule, advanced or otherwise.

The Examiner's response states that appellants are using an interpretation of the phrase, "occurrence of a predetermined event" that is too limiting. Specifically, Examiner's Response states that "occurrence of a predetermined event" should include a device failure that lasts for a lengthy period of time, and that under such an

interpretation the stop event request of Rangarajan would act as an advanced notification rule. Appellants respectfully submit that under Examiner's reading, in which predetermined events last for a lengthy period of time, Rangarajan still fails to disclose an advanced notification rule that preempts a standard notification rule, as recited in claim 25.

The Examiner's Response provides an example to illustrate Examiner's assertion that a stop event request is the same as an advanced notification rule. Specifically, the Examiner's Response states:

> A first event request record (i.e., the standard notification rule) would start at 3:00 p.m. and would send a notification if the device failed (see start time field, threshold one, and relation one of column 7). A second event request record (i.e., the advanced notification rule) would start earlier at 1:00 p.m. and would stop all polling if the device failed in the CPU usage is high (see start time field, both thresholds, and both thresholds of column 7). If at 2:00 PM a device had high CPU usage and failed, the advanced notification rule will trigger and send a stop event.
>
> Such a stop event would preempt the standard notification rule by suspending the first notification rule from being generated upon the occurrence such that the first notification rule is not generated. That is, when 3:00 PM arrived the standard notification rule would not poll because of the stop event and the first notification would not be generated.

(Examiner's Response, page 9).

Appellants respectfully disagree with Examiner's characterization of Rangarajan and submit that Examiner's example does not accurately reflect what is disclosed in Rangarajan. In the Examiner's example, a stop event request disables all subsequent event requests from being performed. However, the stop request is described by Rangarajan in relation to a single specific event request. Each event request includes parameters that determine when, how often, and at what intervals a device will be polled in response to the event request. (Rangarajan, col. 7, lines 19-34). Unless a

stop event request is issued for an event request, that event request will cause a low-level agent to be continually polled for the designated time period. If the stop request field is checked off however, polling of an attribute is stopped for that event request when the monitored attribute satisfies a specified condition. (Rangarajan, col. 7, lines 60-62, col. 8, lines 43-48). As disclosed by Rangarajan, this reduces unnecessary network traffic.

Rangarajan further states that, "once the alternate proxy is delegated, the previous mid-level manager is disengaged by sending a stop event request to the previous mid-level manager." If the stop event request were to stop polling of an attribute for all event requests, then an administrator would have to manually re-enable an attribute every time an alternate proxy is delegated. This would introduce an unnecessary burden to the administrator. Therefore, stop event requests only disable polling of an attribute for a specific event request. Accordingly, if the stop event request were to preempt a notification to suspend a signaling action from occurring, no signaling action would ever occur for the event. As previously stated by Appellants, this would result in an effectively non-functional system.

Referring to the Examiner's example, if at 2:00PM a stop event request is triggered, this would stop polling of the device by only the second event request. The first event request would still begin polling the device at 3:00 PM. Presumably, the first event request would include its own conditions for sending a stop event request. Therefore, if the device has failed as in the Examiner's answer, the first event request would poll the device only briefly, trigger a stop event request for the first event request, and cease polling the device. The stop event request of the second event record would not

preempt the first event record from polling the device, and therefore would not preempt a signaling action generated by the first even record. Accordingly, the stop event request of Rangarajan is not the same as an advanced notification rule.

The Office Action of October 16, 2007 also cited col. 5, lines 56-63 of Rangarajan as disclosing, "enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated." The cited passage is reproduced below:

> The council procedure 70 includes an object-oriented graphical user interface (GUI) for modifying the event request and attributes records in the runtime database 62. The GUI can be derived from open windows 3.1 or later or any other library of classes for GUIs. To modify a record, the console procedure 70 displays the fields for an event request (the fields of an event request record and one or more attributes records can be combined into a single display) and allows the network administrator to fill in or change the fields. The modified records are saved, and the console procedure 70 is restarted.
>
> Reference is now made to FIG. 6, which shows the steps performed by a network administrator while using the console procedure 70. First, the event dispatcher 68 is run in the background (step 200) and then the console procedure 70 is executed (step 202). Upon execution, the console procedure 70 registers with the event dispatcher 68, informing the event dispatcher 68 to forward event reports 78 to it.
>
> If any of the records in the runtime database 62 need to be modified (step 204), the network administrator modifies and saves the records in the runtime database 62 (step 206). The console procedure 70 begins firing event requests 82 at their scheduled start times.
>
> If an event is generated (step 208), the network administrator can view the corresponding event reports 78 via the console procedure 70 (step 210). The event report 78 indicates the attribute and conditions for which the event was generated, the course of action taken by the network manager 48, and the results (if any) from actions taken.
>
> If the event report 78 indicates that the device was down because CPU usage was too high, or because a router on the path was not operational, the network administrator can take the appropriate actions (step

212). If polling of the device attribute has been stopped, no remedial further event reports 78 will be generated for the device.

Thus disclosed is an invention that reduces network management traffic and performs troubleshooting automatically and conveniently from a remote location. The invention greatly reduces the burden of managing a network.

(Rangarajan, col. 5, lines 56-63).

Although it is unclear what language of the cited passage of Rangarajan the Examiner is purporting to disclose an advanced notification rule and a first notification, it appears that the Examiner may be attempting to interpret Rangarajan's disclosure of an event request as an advanced notification rule and Rangarajan's disclosure of an event report as the first notification. It is respectfully submitted that such an interpretation is inapposite.

An event request as described by Rangarajan is not the same as an advanced notification rule. Nor is an event report as described by Rangarajan the same as a notification. Rangarajan defines an event request as a message sent from a network manager to a mid-level manager that directs the mid-level manager to poll a device during a prescribed interval to ascertain an attribute of the device against one or more conditions, and an event report as a message forwarded to the network manager by the mid-level manager when the one or more conditions occur. (Rangarajan, col. 3, lines 32-37). The event request does not include any rules that identify when or how to notify a system administrator or other user when a condition occurs. Nor does the event report include any notification to a system administrator or other user that the condition has occurred. Such a notification is instead accomplished by a signaling action performed by the network manager, the signaling action being distinct from the event

request and the event report. (Rangarajan, col. 5, lines 53-56). However, Rangarajan fails to disclose any rules that control when or how to perform a signaling action. In contrast, claim 25 recites, "a processor configured to enable an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated."

Rangarajan fails to disclose all of the features of claim 25. Accordingly, independent claim 25 and dependent claims 26, 29 and 44 are not anticipated by Rangarajan.

6. Claim 25 and associated dependent claims 26, 29 and 44 are not anticipated by Rangarajan because Rangarajan fails to disclose a processor configured to enable a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy.

Appellants respectfully submit that Rangarajan does not disclose a processor configured to enable a notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy. Rangarajan discloses a network manager that performs a signaling action (e.g., sounds an alarm) upon receiving event reports. (Rangarajan, col. 5, lines 39-56). The signaling action notifies a network administrator of the event. (Rangarajan, col. 5, lines 53-56). However, Rangarajan does not disclose multiple different persons who can be notified of an event. Nor does Rangarajan disclose a hierarchy of persons to notify of the event. Rangarajan also does not disclose any rules that determine under what circumstances a particular person should be notified of an event. In contrast, claim 25 recites, "a

processor configured to enable a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy."

Rangarajan fails to disclose all of the features of claim 25. Accordingly, independent claim 25 and dependent claims 26, 29 and 44 are not anticipated by Rangarajan.

7. Claims 8, 16 and 24 are not anticipated by Rangarajan because Rangarajan fails to disclose an advanced notification rule configured to preempt a standard notification rule for a temporary amount of time.

As discussed above with reference to claims 1 and 9, Rangarajan fails to disclose enabling an advanced notification rule to preempt a standard notification rule. Moreover, Rangarajan also fails to disclose any conditions that apply to standard notification rules or to advanced notification rules. Therefore, Rangarajan does not disclose an advanced notification rule configured to preempt a standard notification rule for a temporary amount of time, as recited in claims 8, 16 and 24.

The Examiner cites col. 7, lines 1-38 as disclosing an advanced notification rule configured to preempt a standard notification rule for a temporary amount of time. The cited passage describes a start time and stop time for checking an attribute of a device identified in an event request. (Rangarajan, col. 7, lines 28-30). However, as established above, the event request is not an advanced notification rule. Nor does Rangarajan describe the event request as preempting another event request, much less

as preempting another event request for a temporary amount of time. Accordingly, claims 8, 16, and 24 are not anticipated by Rangarajan.

B. Claims 2, 10, 18 and 26 are not rendered obvious by the combination of Rangarajan and Graf because neither Rangarajan nor Graf teach all of the features of these claims.

1. Claims 2 and 10 are not rendered obvious by the combination of Rangarajan and Graf because neither Rangarajan nor Graf teach all of the features of these claims.

As discussed above with reference to claim 1 and 9, Rangarajan fails to disclose enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated.

Graf teaches a network monitoring system that generates an alert when predetermined conditions are met. (Graf, col. 19, line 38 to col. 22, line 16). The alert can include a notify property that identifies a notification action to take when the alert is generated. (Graf, Table 13, ALERT: NOTIFY; table 14, setNotify and doNotify). The alert can time out (Graf, col. 20, lines 1-4), it can be cleared (Graf, col. 20, lines 39-49) or it can be ignored (Graf, col. 20, line 50 to col. 21, line 8). However, Graf does not teach that the alert can be preempted. Moreover, the acts of timing out, clearing and ignoring the alert are all performed in response to input received by a system administrator or automatically based on parameters of the alert itself. None of these actions are achieved based on the contents of a different alert (e.g., of an advanced

alert). Nor does Graf teach enabling an advanced alert to preempt a standard alert by suspending a notification of the standard alert from being generated upon the occurrence of a condition that caused the standard alert. Accordingly, Graf fails to teach the features of independent claim 1 missing from Rangarajan.

Neither Rangarajan nor Graf, alone or in combination, teach or suggest all of the limitations of independent claims 1 or 9. Claim 2 depends from claim 1, and is therefore patentable for at least the reasons that claim 1 is patentable. Claim 10 depends from claim 9, and is therefore patentable for at least the reasons that claim 9 is patentable.

2. Claim 18 is not rendered obvious by the combination of Rangarajan and Graf because neither Rangarajan nor Graf teach all of the features of claim 18.

As discussed above with reference to claim 17, Rangarajan fails to disclose means for enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated.

Graf teaches a network monitoring system that generates an alert when predetermined conditions are met. (Graf, col. 19, line 38 to col. 222, line 16). The alert can include a notify property that identifies a notification action to take when the alert is generated. (Graf, Table 13, ALERT: NOTIFY; table 14, setNotify and doNotify). The alert can time out (Graf, col. 20, lines 1-4), it can be cleared (Graf, col. 20, lines 39-49) or it can be ignored (Graf, col. 20, line 50 to col. 21, line 8). However, Graf does not teach that the alert can be preempted. Moreover, the acts of timing out, clearing and ignoring the alert are all performed in response to input received by a system

administrator or automatically based on parameters of the alert itself. None of these actions are achieved based on the contents of a different alert (e.g., of an advanced alert). Nor does Graf teach enabling an advanced alert to preempt a standard alert by suspending a notification of the standard alert from being generated upon the occurrence of a condition that caused the standard alert. Accordingly, Graf fails to teach the features of independent claim 17 missing from Rangarajan.

Neither Rangarajan nor Graf, alone or in combination, teach or suggest all of the limitations of independent claim 17. Claim 18 depends from claim 17, and is therefore patentable for at least the reasons that claim 17 is patentable.

3. Claim 26 is not rendered obvious by the combination of Rangarajan and Graf because neither Rangarajan nor Graf teach all of the features of claim 26.

As discussed above with reference to claim 25, Rangarajan fails to disclose a processor configured to enable an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated.

Graf teaches a network monitoring system that generates an alert when predetermined conditions are met. (Graf, col. 19, line 38 to col. 222, line 16). The alert can include a notify property that identifies a notification action to take when the alert is generated. (Graf, Table 13, ALERT: NOTIFY; table 14, setNotify and doNotify). The alert can time out (Graf, col. 20, lines 1-4), it can be cleared (Graf, col. 20, lines 39-49) or it can be ignored (Graf, col. 20, line 50 to col. 21, line 8). However, Graf does not teach that the alert can be preempted. Moreover, the acts of timing out, clearing and

ignoring the alert are all performed in response to input received by a system administrator or automatically based on parameters of the alert itself. None of these actions are achieved based on the contents of a different alert (e.g., of an advanced alert). Nor does Graf teach enabling an advanced alert to preempt a standard alert by suspending a notification of the standard alert from being generated upon the occurrence of a condition that caused the standard alert. Accordingly, Graf fails to teach the features of independent claim 25 missing from Rangarajan.

Neither Rangarajan nor Graf, alone or in combination, teach or suggest all of the limitations of independent claim 25. Claim 26 depends from claim 25, and is therefore patentable for at least the reasons that claim 25 is patentable.

VIII. CONCLUSION

Based on the foregoing, Appellants respectfully submit that the Board should reverse the rejections of all pending claims and hold that all of the claims currently under review are allowable.

Respectfully submitted,

Dated: 6/30, 2008

Daniel E. Ovanezian
Reg. No. 41,236

Customer No. 066701

IX. CLAIMS APPENDIX

The claims involved in this appeal are presented below.

1. (Previously Presented) A method, comprising:

enabling a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy; and

enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated.

2. (Previously Presented) The method of claim 1 further comprising:

generating a second notification to a second person in the hierarchy based on the advanced notification rule.

3. (Canceled)

4. (Canceled)

5. (Canceled)

6. (Canceled)

7. (Previously Presented) The method of claim 1, wherein the advanced notification rule includes a scope and wherein the scope of the advanced notification rule is configured by at least one of the group consisting of a company, a satellite, a host assigned to a company, a service configured on a host for a company, a check type, a host state, a service state, a contact group, and a message pattern.

8. (Previously Presented) The method of claim 1 where the advanced notification rule is configured to preempt the standard notification rule for a temporary amount of time.

9. (Previously Presented) A machine readable medium having stored thereon instructions, which when executed by a processor, cause the processor to perform the following:

enabling a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy; and

enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated.

10. (Original) The machine readable medium of claim 9 further comprising:

generating a second notification to a second person in the hierarchy.

11. (Canceled)

12. (Canceled)

13. (Canceled)

14. (Canceled)

15. (Previously Presented) The machine readable medium of claim 9, wherein the advanced notification rule includes a scope where the scope of the advanced

notification rule configured by at least one of the group consisting of a company, a satellite, a host assigned to a company, a service configured on a host for a company, a check type, a host state, a service state, a contact group, and a message pattern.

16. (Previously Presented) The machine readable medium of claim 9, wherein the advanced notification rule is configured to preempt the standard notification rule for a temporary amount of time.

17. (Previously Presented) An apparatus, comprising:

means for enabling a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy; and

means for enabling an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated.

18. (Original) The apparatus of claim 17 further comprising:

means for generating a second notification to a second person in the hierarchy.

19. (Canceled)

20. (Canceled)

21. (Canceled)

22. (Canceled)

23. (Previously Presented) The apparatus of claim 17, wherein the advanced notification rule includes a scope and wherein the scope of the advanced notification rule is configured by at least one of the group consisting of a company, a satellite, a host assigned to a company, a service configured on a host for a company, a check type, a host state, a service state, a contact group, and a message pattern.

24. (Previously Presented) The apparatus of claim 17 where the advanced notification rule is configured to preempt the standard notification rule for a temporary amount of time.

25. (Previously Presented) An digital processing system, comprising:

a processor configured to enable a standard notification rule to generate a first notification upon an occurrence of a predetermined event to a first person in a hierarchy, and to enable an advanced notification rule to preempt the standard notification rule by suspending the first notification from being generated upon the occurrence such that the first notification is not generated; and

a communications device coupled to the processor to transmit the notifications.

26. (Previously Presented) The digital processing system of claim 25 wherein the communications device is configured to transmit the second notification to a second person in the hierarchy based on the advanced notification rule.

27. (Canceled)

28. (Canceled)

29. (Original) The digital processing system of claim 25 where the communications device transmits the first notification to the first person in the hierarchy and the processor acknowledges the first notification.

30. (Canceled)

31. (Canceled)

32. (Canceled)

33. (Canceled)

34. (Canceled)

35. (Canceled)

36. (Canceled)

37. (Canceled)

38. (Canceled)

39. (Canceled)

40. (Canceled)

41. (Previously Presented) The method of claim 1, wherein the advanced notification rule is enabled to preempt the standard notification rule while continuing monitoring for the predetermined event.

42. (Previously Presented) The machine readable medium of claim 9, wherein the advanced notification rule is enabled to preempt the standard notification rule while continuing monitoring for the predetermined event.

43. (Previously Presented) The apparatus of claim 17, wherein the advanced notification rule is enabled to preempt the standard notification rule while continuing monitoring for the predetermined event.

44. (Previously Presented) The digital processing system of claim 25, wherein the processor is configured to enable the advanced notification rule to preempt the standard notification rule while continuing monitoring for the predetermined event.

X. EVIDENCE APPENDIX

No other evidence is submitted in connection with this appeal.

XI. RELATED PROCEEDINGS APPENDIX

No related proceedings exist.